s10

Abigail Ramos

4/11/2022

```
## Loading required package: splines
## Loading required package: RcmdrMisc
## Loading required package: car
## Loading required package: carData
## Loading required package: sandwich
## Loading required package: effects
## lattice theme set by effectsTheme()
## See ?effectsTheme for details.
## La interfaz R-Commander sólo funciona en sesiones interactivas
##
## Attaching package: 'Rcmdr'
## The following object is masked from 'package:base':
##
##     errorCondition
```

REGRESIÓN LINEAL SIMPLE

- EJEMPLO 1.

En el archivo "costes.dat" se encuentra la información correspondiente a 34 fábricas de producción en el montaje de placas para ordenador, el archivo contiene la información sobre el costo total (primera columna) y el número de unidades fabricadas (segunda columna). Suponga que deseamos ajustar un modelo de regresión simple a los datos para estimar el costo total en función del número de unidades fabricadas.

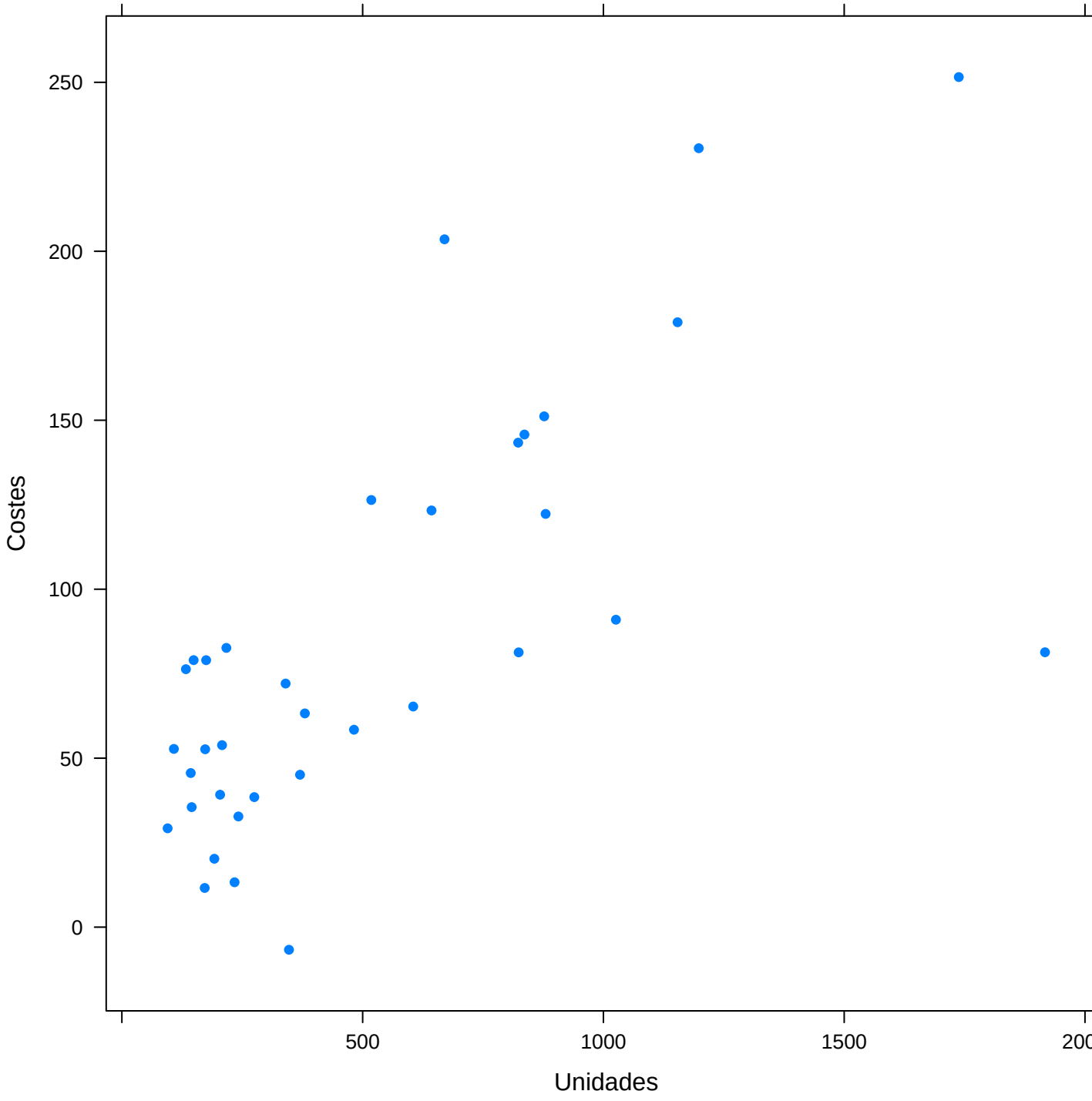
```
> costes <- read.table("C:/Users/abbyc/Desktop/Ciclo II 2022/Análisis estadístico con R/curso-R-2022/co
+                      header=TRUE, stringsAsFactors=TRUE, sep="", na.strings="NA",
+                      dec=".", strip.white=TRUE)
```

Lo primero que debemos es hacer es graficar los datos. Para obtener el diagrama de dispersión de las variables el procedimiento es el siguiente: en el menú "Gráficas" seleccionar la opción "Gráfica XY".

Al realizar el procedimiento anterior se mostrará un cuadro de dialogo como el de la figura siguiente. En el únicamente debemos elegir las variables que se graficarán. En el recuadro de la parte derecha debemos seleccionar a nuestra variable independiente, la cual hemos dicho que es el número de unidades producidas; mientras que en el recuadro de la derecha debemos elegir nuestra variable dependiente, que para nuestro ejemplo es el costo total. Los demás argumentos se dejan por defecto.

```
> library(lattice)
> xyplot(Costes ~ Unidades, type="p", pch=16, auto.key=list(border=TRUE),
```

```
+   par.settings=simpleTheme(pch=16), scales=list(x=list(relation='same'),
+   y=list(relation='same')), data=costes)
```



Para ajustar un modelo de regresión lineal en la interfaz gráfica de R, el procedimiento es el siguiente: en el menú "Estadísticos" seleccionamos la opción "Ajuste de modelos", finalmente debemos elegir la opción

“Regresión lineal”.

Al realizar el procedimiento descrito anteriormente nos mostrará un cuadro de dialogo en el que debemos tener en cuenta lo siguiente: el nombre que le daremos al modelo de regresión resultante, este nombre se da en la opción “Introducir un nombre para el modelo”. En el recuadro de la izquierda debemos seleccionar a nuestro variable dependiente (Costos); mientras que en el recuadro de la derecha debemos seleccionar a nuestra variable independiente (Unidades).

```
> RegModel.1 <- lm(Costes~Unidades, data=costes)
> summary(RegModel.1)

Call:
lm(formula = Costes ~ Unidades, data = costes)

Residuals:
     Min       1Q   Median       3Q      Max
-137.386  -24.496   -0.117   29.848  105.028

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 33.92200   11.57500   2.931   0.0061 **
Unidades     0.09640    0.01665   5.789  1.8e-06 ***
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1

Residual standard error: 44.49 on 33 degrees of freedom
Multiple R-squared:  0.5039,    Adjusted R-squared:  0.4888
F-statistic: 33.51 on 1 and 33 DF,  p-value: 1.796e-06
```

Como el término constante no es significativo se quitara del modelo, volvemos a realizar los cálculos en la interfaz gráfica. En el menú “Estadísticos” seleccionamos la opción “Ajuste de modelos” y finalmente la opción “Modelo lineal”. Esta opción nos permite descartar la constante del modelo (debemos agregar -1 al final de la instrucción).

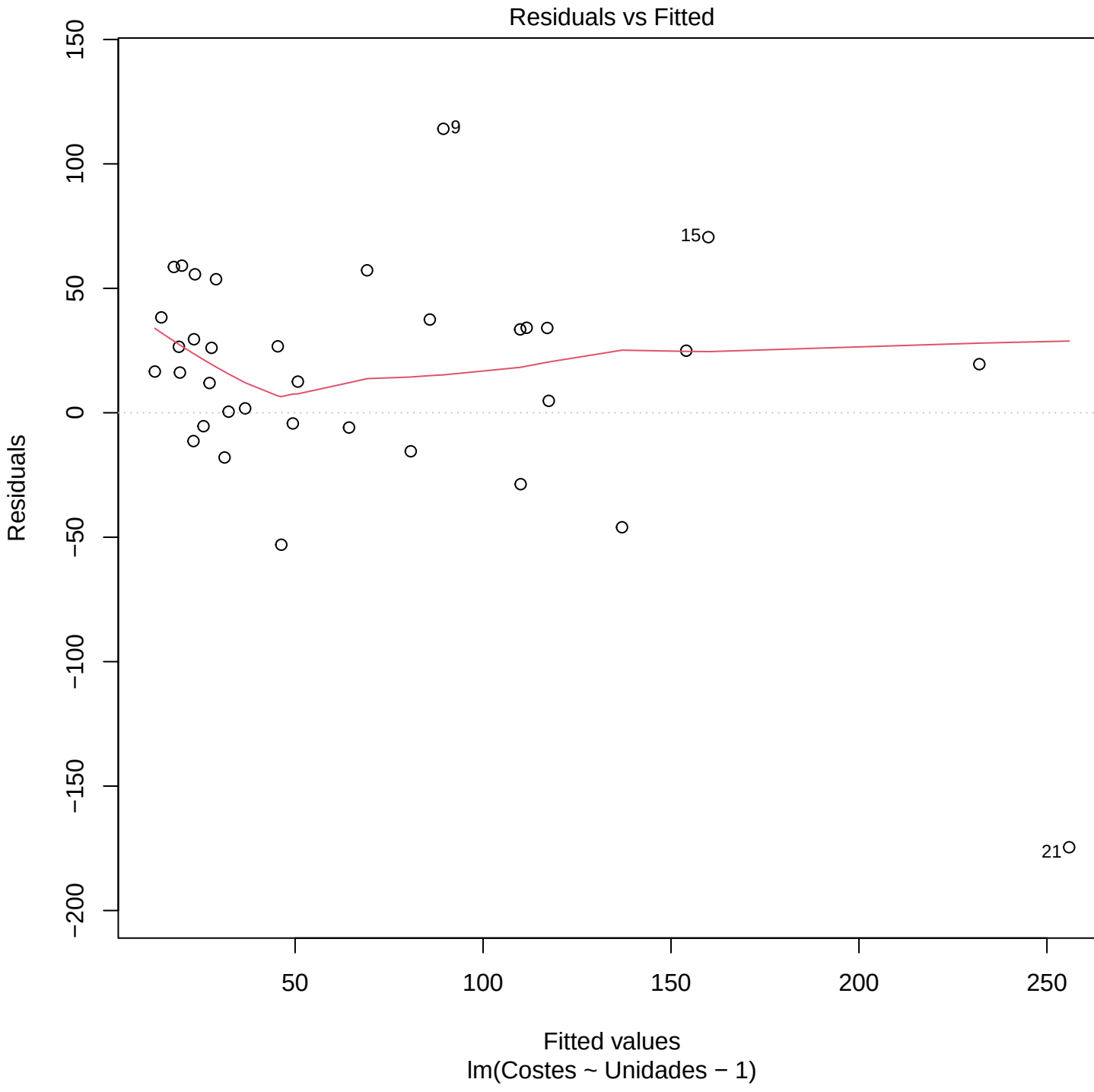
```
> LinearModel.2 <- lm(Costes ~ Unidades-1, data=costes)
> summary(LinearModel.2)

Call:
lm(formula = Costes ~ Unidades - 1, data = costes)

Residuals:
     Min       1Q   Median       3Q      Max
-174.579   -4.844   19.527   35.812  114.095

Coefficients:
         Estimate Std. Error t value Pr(>|t|)
Unidades  0.13350    0.01197   11.16 6.59e-13 ***
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1

Residual standard error: 49.21 on 34 degrees of freedom
Multiple R-squared:  0.7854,    Adjusted R-squared:  0.7791
F-statistic: 124.5 on 1 and 34 DF,  p-value: 6.591e-13
```

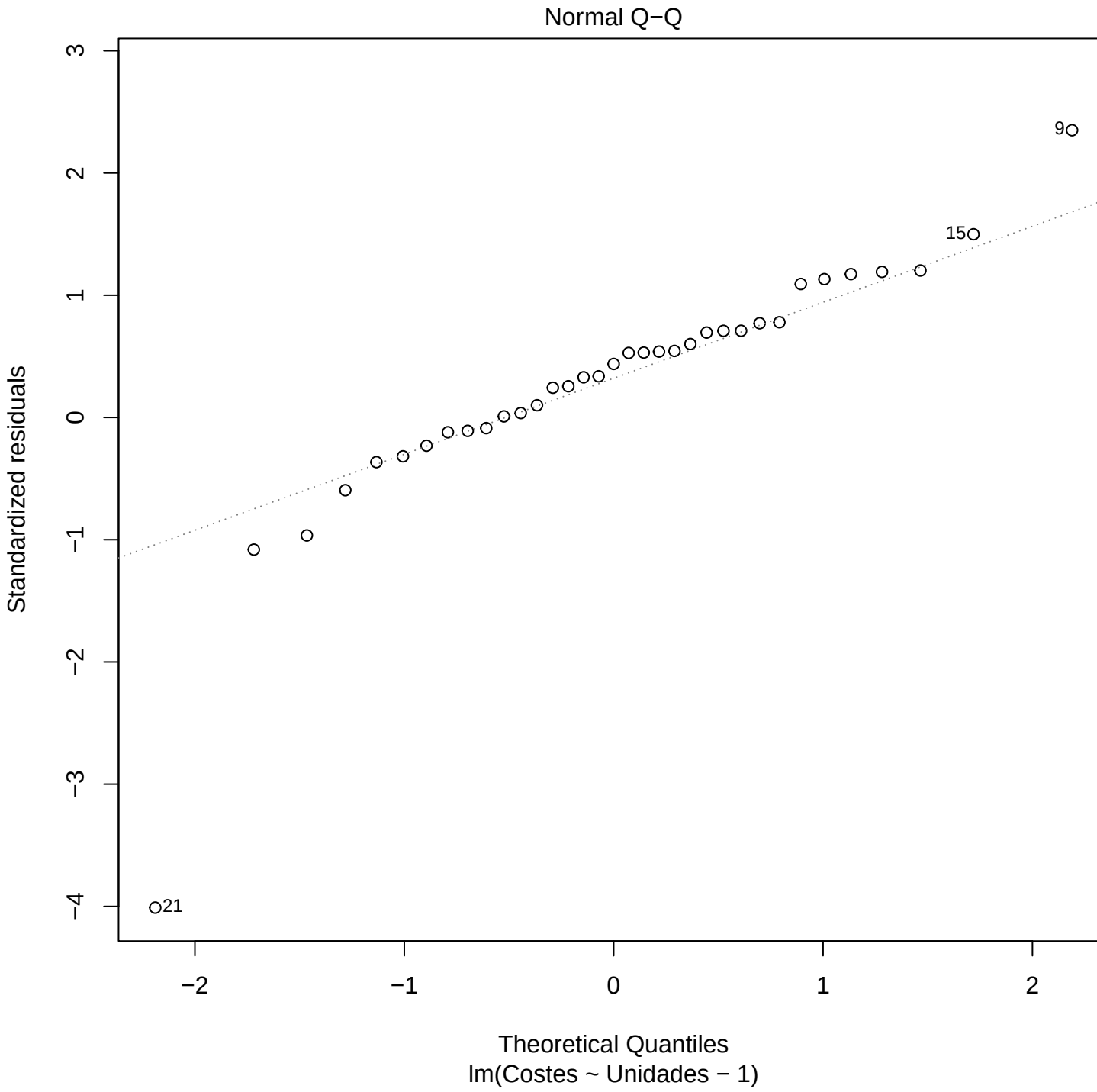
Una vez estimados los parámetros del modelo, el siguiente paso es validarlo, es decir verificar si se cumplen las cuatro hipótesis básicas del modelo. Para verificar esto, podríamos realizar los siguientes pasos:

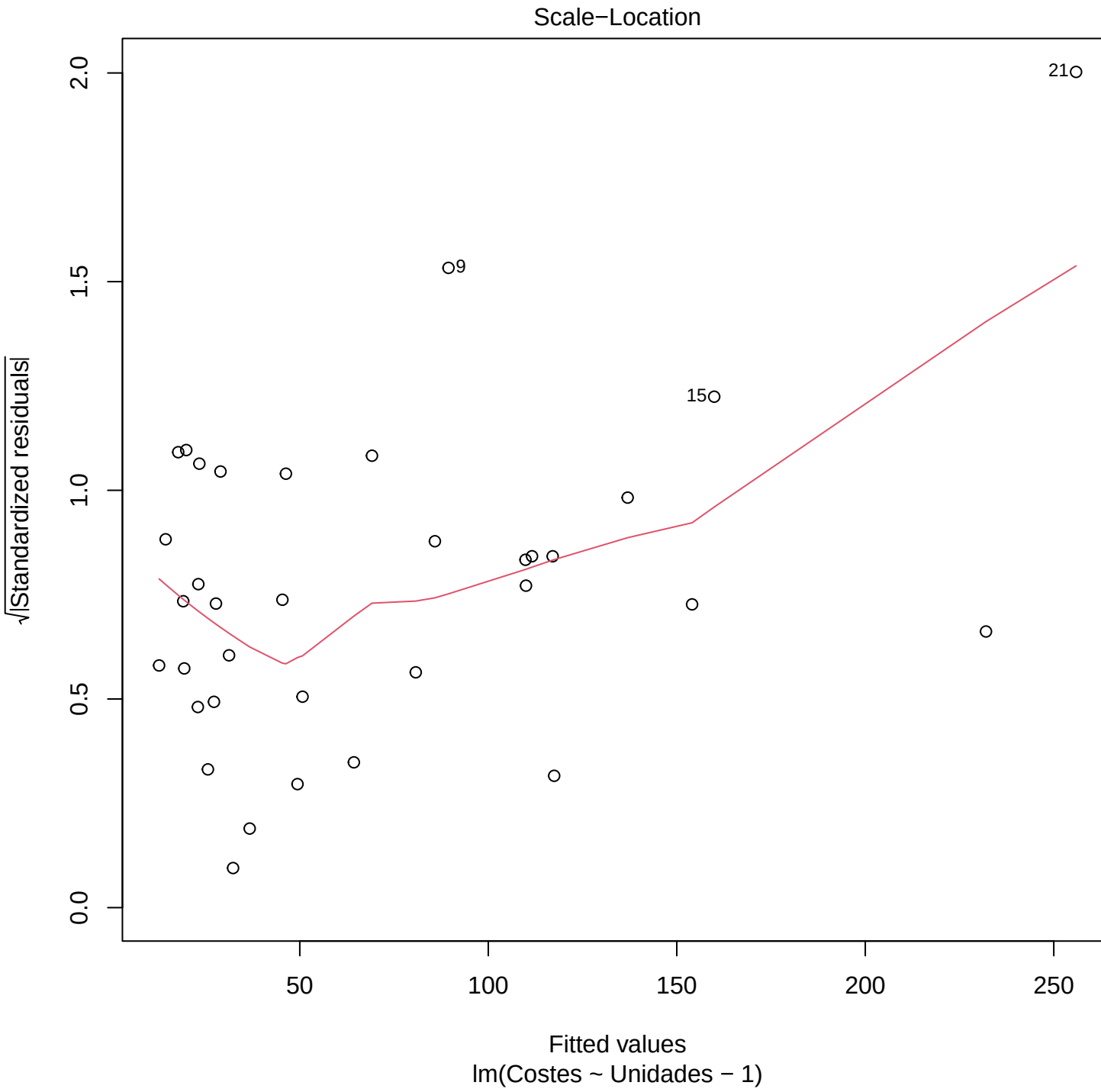
En el menú “Modelos” seleccionamos la opción “Gráficas”, posteriormente seleccionamos la opción “Gráficas básicas del modelo”.

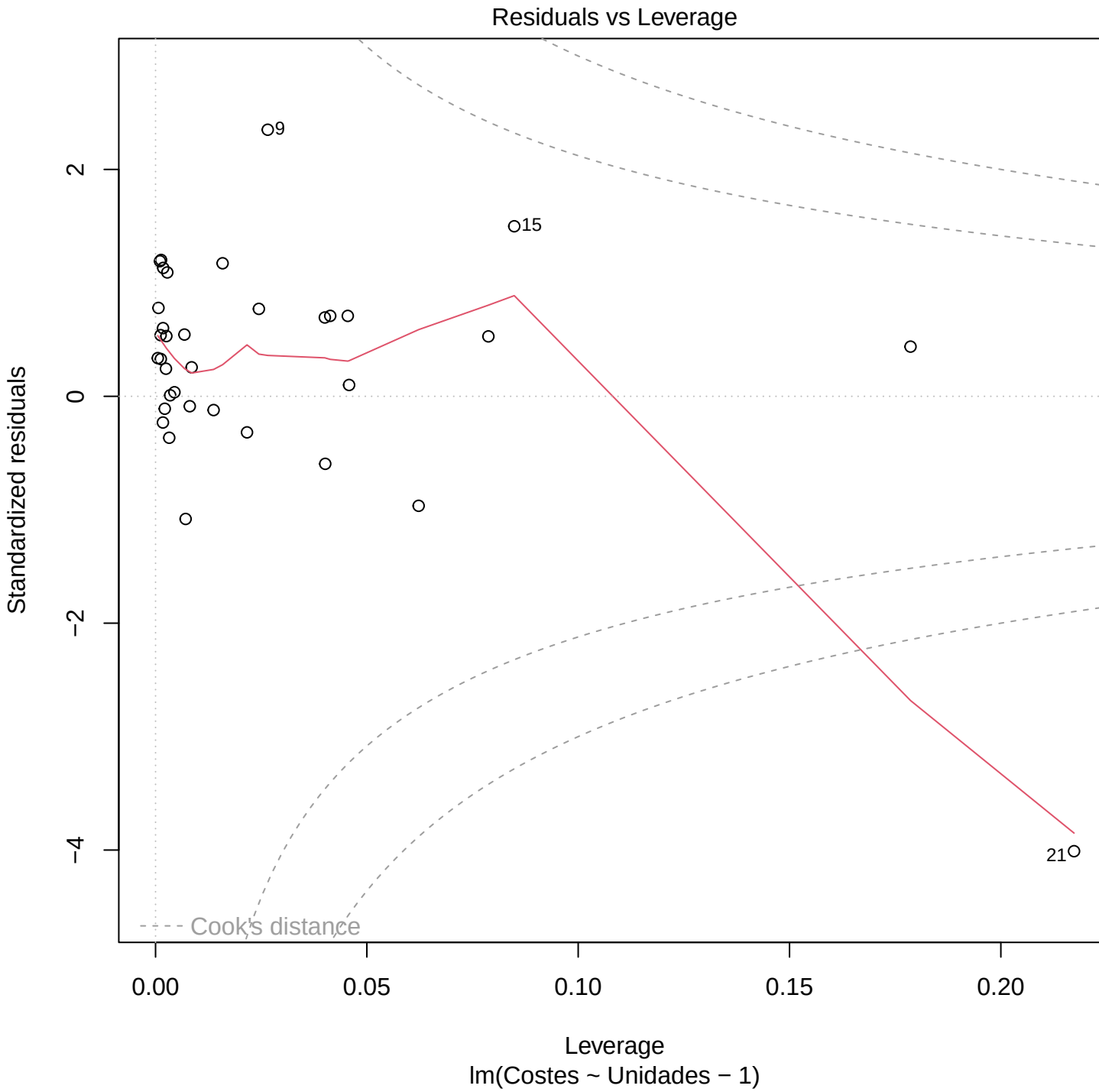
```
> oldpar <- par(oma=c(0,0,3,0), mfrow=c(2,2))
```

```
> plot(LinearModel.2)
```




Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(Costes ~ Unidades − 1)
9
15
21
3
2
1
0
−1
−2
−3
−4
−2
−1
0
1
2


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
21
9
15
50
100
150
200
250
Fitted values
lm(Costes ~ Unidades − 1)




```
> par(oldpar)
```

El procedimiento para obtener las medidas anteriores es el siguiente: en el menú "Modelos" seleccionamos la opción "Añadir las estadísticas de las observaciones a los datos...". Posteriormente en el cuadro de dialogo que se mostrará elegir todas las opciones que se quieran analizar.

```
> costes<- within(costes, {
+   fitted.LinearModel.2 <- fitted(LinearModel.2)
+   residuals.LinearModel.2 <- residuals(LinearModel.2)
+   rstudent.LinearModel.2 <- rstudent(LinearModel.2)
+   hatvalues.LinearModel.2 <- hatvalues(LinearModel.2)
+   cooks.distance.LinearModel.2 <- cooks.distance(LinearModel.2)
+ })
```

REGRESIÓN LINEAL MÚLTIPLE

- EJEMPLO 2.

En el archivo "preciocasas.dat" tienen la información sobre 100 datos de precios de viviendas y sus características, el archivo se encuentra estructurado de la siguiente forma:

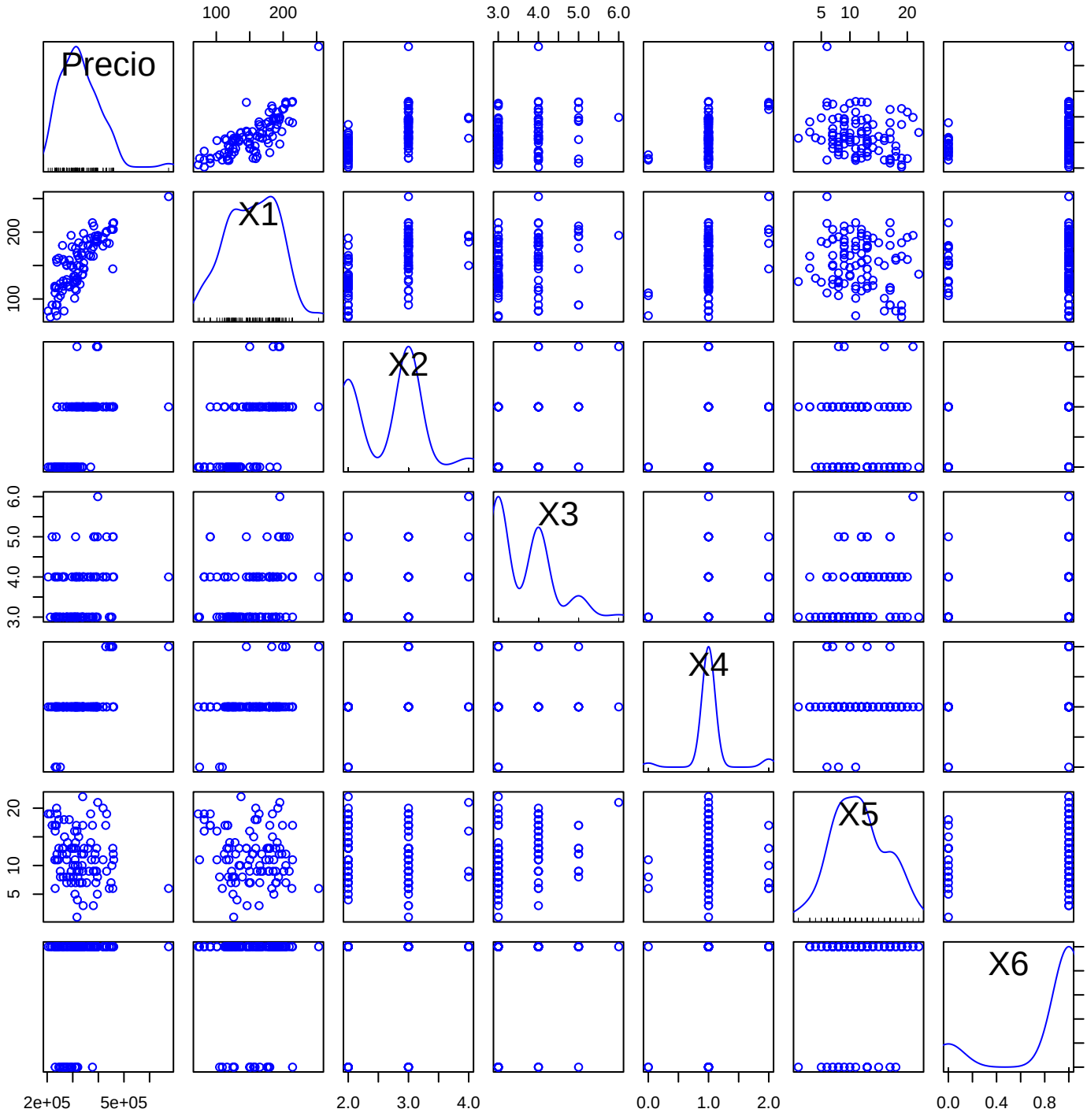
```
> preciocasas <-
+   read.table("C:/Users/abbyc/Desktop/Ciclo II 2022/Análisis estadístico con R/curso-R-2022/preciocasas
+    header=TRUE, stringsAsFactors=TRUE, sep="", na.strings="NA", dec=".",
+   strip.white=TRUE)
> names(preciocasas)<-c("Precio","X1","X2","X3","X4","X5","X6")
```

- Primera columna: precios de viviendas en euros.
- Segunda columna: superficie en metros cuadrados.
- Tercera: numero de cuartos de baño.
- Cuarta: número de dormitorios.
- Quinta: número de plazas de garaje.
- Sexta: edad de la vivienda .
- Séptima: 1 =buenas vistas y 0 =vistas corrientes

Suponga que deseamos estimar un modelo de regresión en el cual relacionemos el precio de una vivienda en función de sus características.

Lo primero que debemos hacer es la matriz de diagramas de dispersión. El procedimiento para obtenerla es el siguiente: en el menú "Gráficas" seleccionamos la opción "Matriz de diagramas de dispersión. . . ".

```
> library(car)
> scatterplotMatrix(~Precio+X1+X2+X3+X4+X5+X6, regLine=FALSE, smooth=FALSE, diagonal=list(method="densi
```



Para ajustar el modelo de regresión múltiple el procedimiento es el siguiente: en el menú "Estadísticos" seleccionamos la opción "Ajuste de modelos", finalmente elegimos la opción "Regresión lineal".

Al realizar el procedimiento anterior nos mostrará un cuadro de dialogo como el de la figura siguiente. En el recuadro de la izquierda debemos seleccionar nuestra variable dependiente (Precio), mientras que el recuadro de la derecha debemos todas las variables independientes (todas las restantes variables).

```
> RegModel.3 <- lm(Precio~X1+X2+X3+X4+X5+X6, data=preciocasas)
> summary(RegModel.3)

Call:
lm(formula = Precio ~ X1 + X2 + X3 + X4 + X5 + X6, data = preciocasas)

Residuals:
    Min      1Q  Median      3Q     Max
-100642  -22619    -332   16951  140920

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  34071.8    26304.5   1.295  0.19846
X1            1175.3      142.2   8.265 1.01e-12 ***
X2           12006.3     9251.9   1.298  0.19763
X3            8187.1     6665.5   1.228  0.22247
X4           60495.3    14500.1   4.172 6.83e-05 ***
X5           -3086.0      969.6  -3.183  0.00199 **
X6           30798.1    11111.0   2.772  0.00675 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 38620 on 92 degrees of freedom
Multiple R-squared:  0.752, Adjusted R-squared:  0.7358
F-statistic: 46.49 on 6 and 92 DF,  p-value: < 2.2e-16
```

De los resultados anteriores puede apreciarse que el intercepto, y las variables x2 (número de cuarto de baño) y x3 (número de dormitorios) no parecen influir en la estimación del precio de la vivienda por lo podrían descartarse de la ecuación.

Una forma alternativa y mucho más eficiente para seleccionar el mejor conjunto de variables independientes en el modelo es utilizar algoritmos selección de modelos tales como: Selección hacia adelante, selección hacia atrás y Selección paso a paso.

El procedimiento para realizar cualquiera de los algoritmos anteriores es el siguiente: en el menú "Modelos" seleccionamos la opción "Selección de modelos paso a paso. . . ".

Al realizar el procedimiento anterior nos mostrara un cuadro de dialogo como el de la figura siguiente. En dicho cuadro únicamente debemos elegir la dirección del criterio de selección de variables teniendo en cuenta únicamente que: atrás/adelante para una selección por pasos en el que se inicia con todas las variables; adelante/atrás es para una selección por pasos pero iniciando con ninguna variable en el modelo; finalmente las opciones Atrás y Adelante son para la selección hacia atrás y selección hacia adelante, respectivamente. Finalmente lo único que debe tenerse en cuenta es el criterio para seleccionar los modelos los cuales son: el criterio AIC y el BIC, ambos son equivalentes, pero en el segundo se penaliza más el número de variables en el modelo, evitando así obtener un modelo con demasiadas variables.

```
> library(MASS, pos=18)
> #stepwise(RegModel.3, direction='backward/forward', criterion='BIC')
```

Otra cosa que es de tener en cuenta a la hora de seleccionar variables en el modelo es que no exista multicolinealidad, es decir, que no exista dependencia entre las variables independientes. La multicolinealidad se estudia con ayuda del siguiente procedimiento: en el menú "Modelos" seleccionamos la opción "Diagnósticos numéricos", finalmente elegimos la opción "Factores de inflación de la varianza".

```
> vif(RegModel.3)

      X1       X2       X3       X4       X5       X6
```

```
1.876271 1.773771 1.424941 1.255808 1.256000 1.110249
```

```
> round(cov2cor(vcov(RegModel.3)), 3) # Correlations of parameter estimates
```

```
            (Intercept)     X1     X2     X3     X4     X5     X6
(Intercept)       1.000 -0.166 -0.249 -0.365 -0.239 -0.251 -0.096
X1               -0.166  1.000 -0.518 -0.124 -0.343  0.191  0.124
X2               -0.249 -0.518  1.000 -0.280  0.017  0.094 -0.124
X3               -0.365 -0.124 -0.280  1.000  0.047 -0.407 -0.100
X4               -0.239 -0.343  0.017  0.047  1.000 -0.054 -0.226
X5               -0.251  0.191  0.094 -0.407 -0.054  1.000 -0.060
X6               -0.096  0.124 -0.124 -0.100 -0.226 -0.060  1.000
```

Recordar únicamente que se dice que existe multicolinealidad cuando los valores correspondientes de VIF (factor de inflación de varianza) para cada variable sea mayor que 5 (y en tal caso tendría que descartase la variable). Para nuestro caso no tenemos ese problema para ninguna variable.